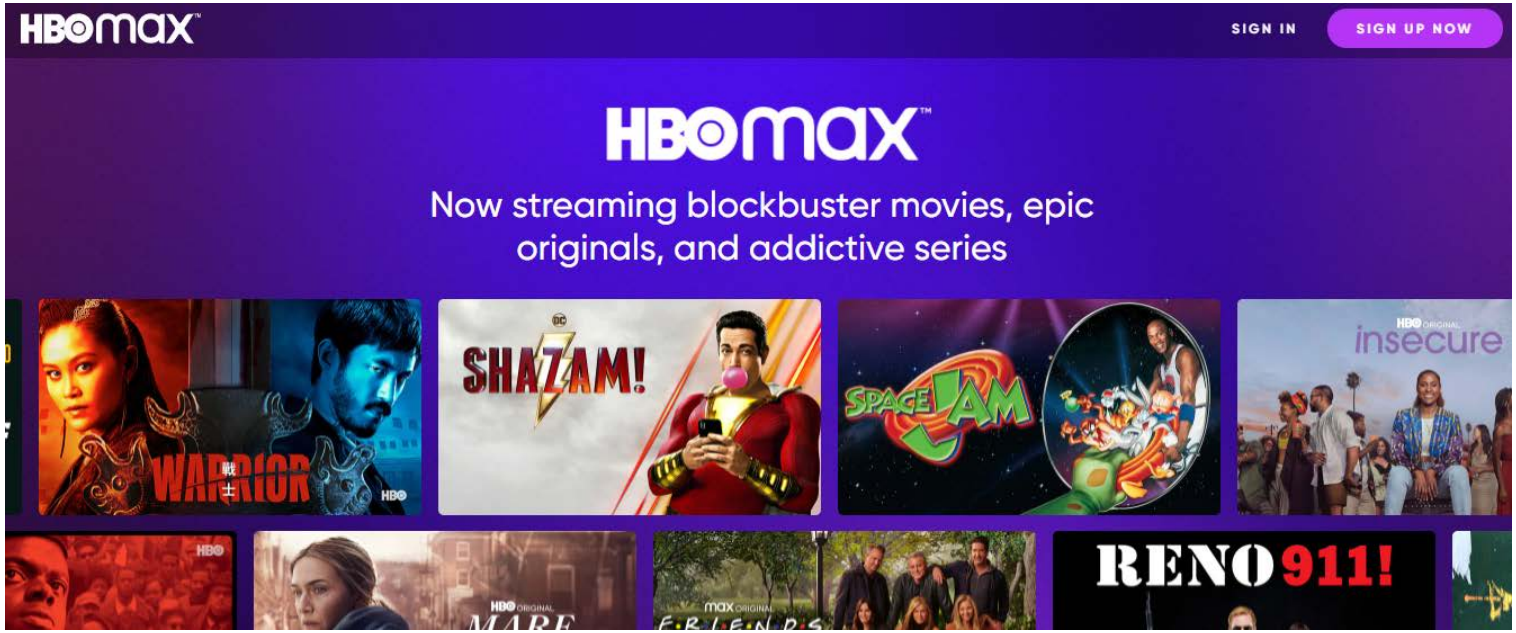
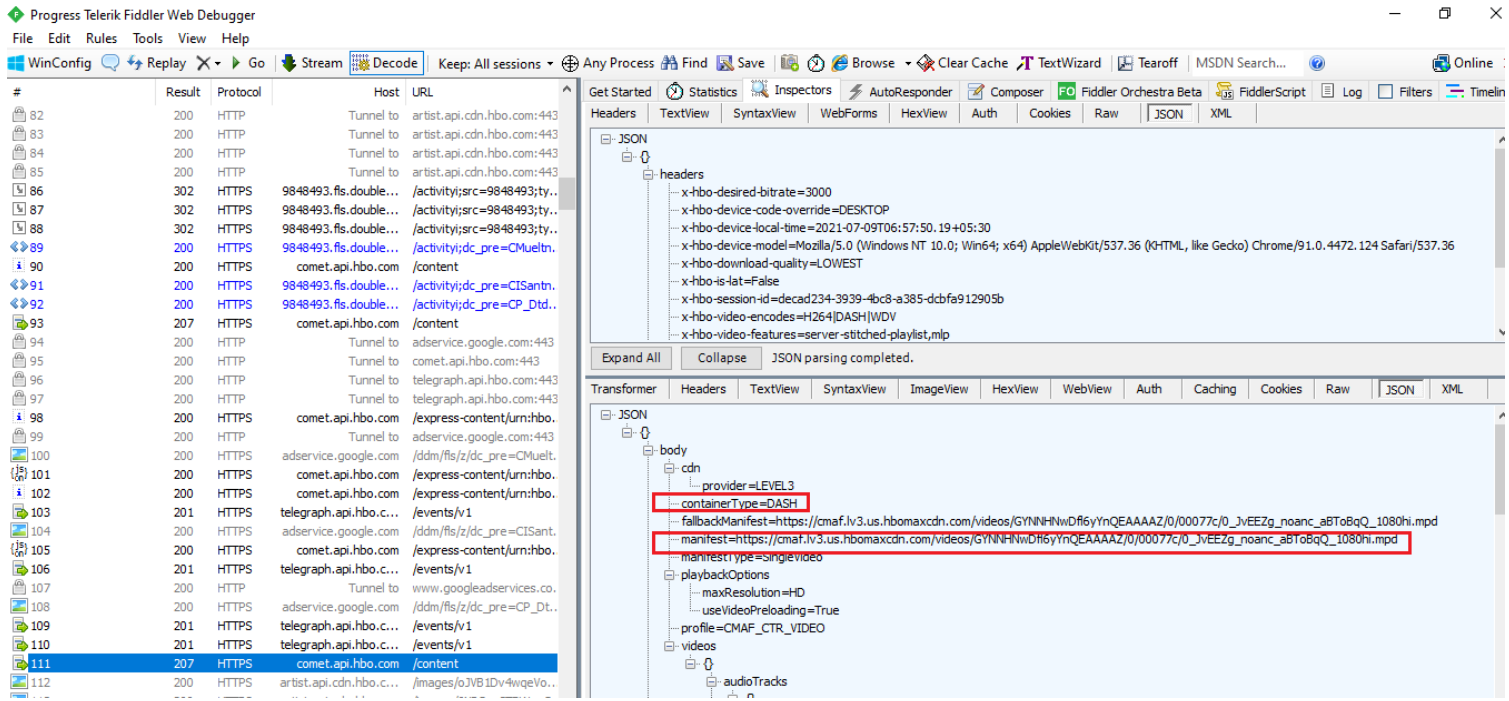
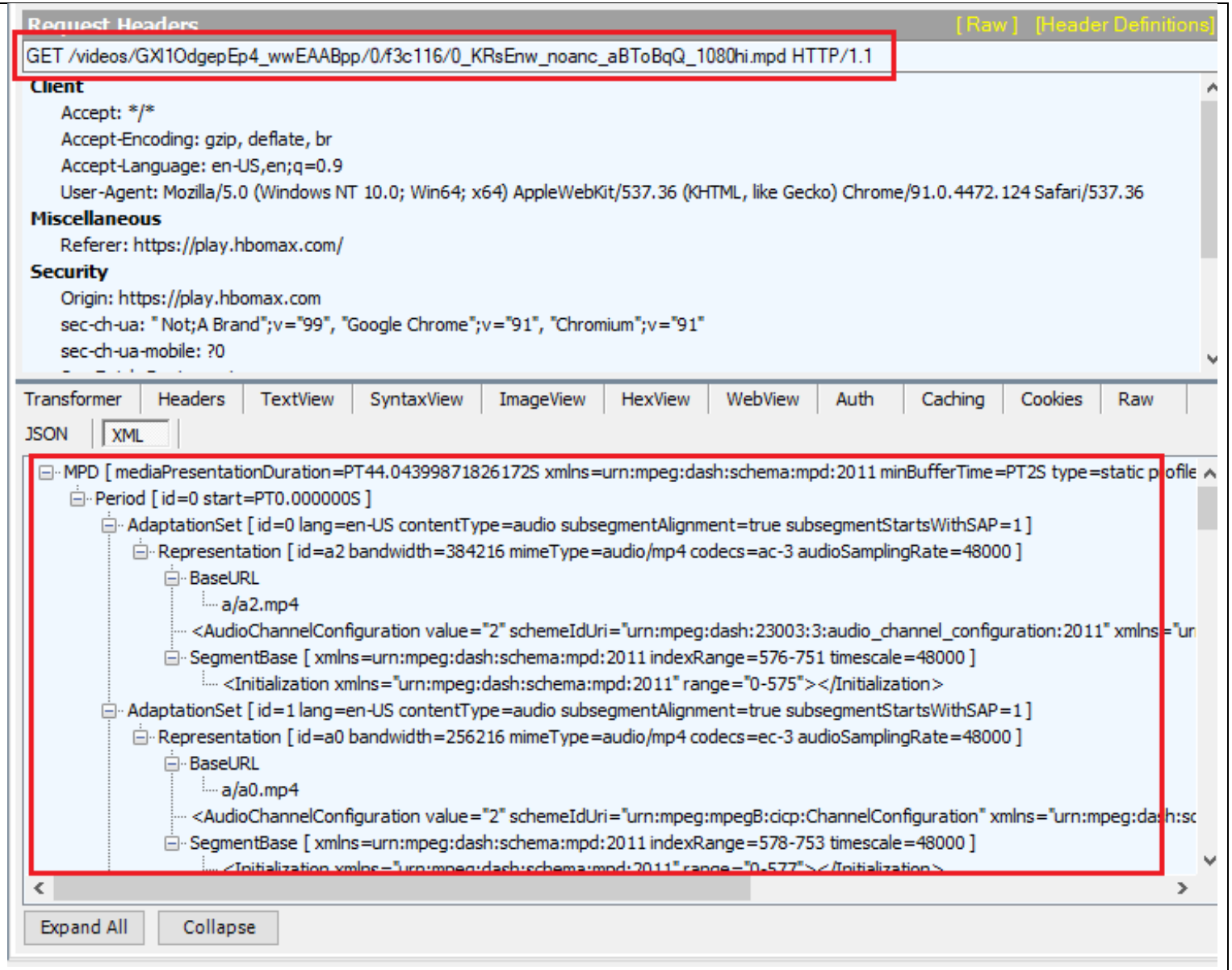
EXHIBIT C

US7079752	HBO
1. A process for recording, on a recording medium, a scrambled digital video stream, implementing the following steps, in addition to the recording of the scrambled data:	HBO utilizes the MPEG-DASH streaming protocol ("the Standard"), including for delivery of VOD contents to its viewers/customers. As shown below, video content from HBO is streamed and the data traffic is captured showing the media format as MPEG-DASH, as indicated by the MPD file (e.g., the Media Presentation Description file is used by MPEG-DASH to contain information about the media playing), and the encryption (e.g., AES 128) scheme used by the streamed video. In addition, HBO provides trick mode operation (such as 15 sec forward/rewind, etc.) to the streamed video. The Standard practices a process for recording (e.g., recording by means of downloading in a storage), on a recording medium (e.g., a storage mechanism), a scrambled digital video stream (e.g., scrambled video created by utilizing AES 128 encryption), implementing the following steps, in addition to the recording of the scrambled data.  https://www.hbomax.com/

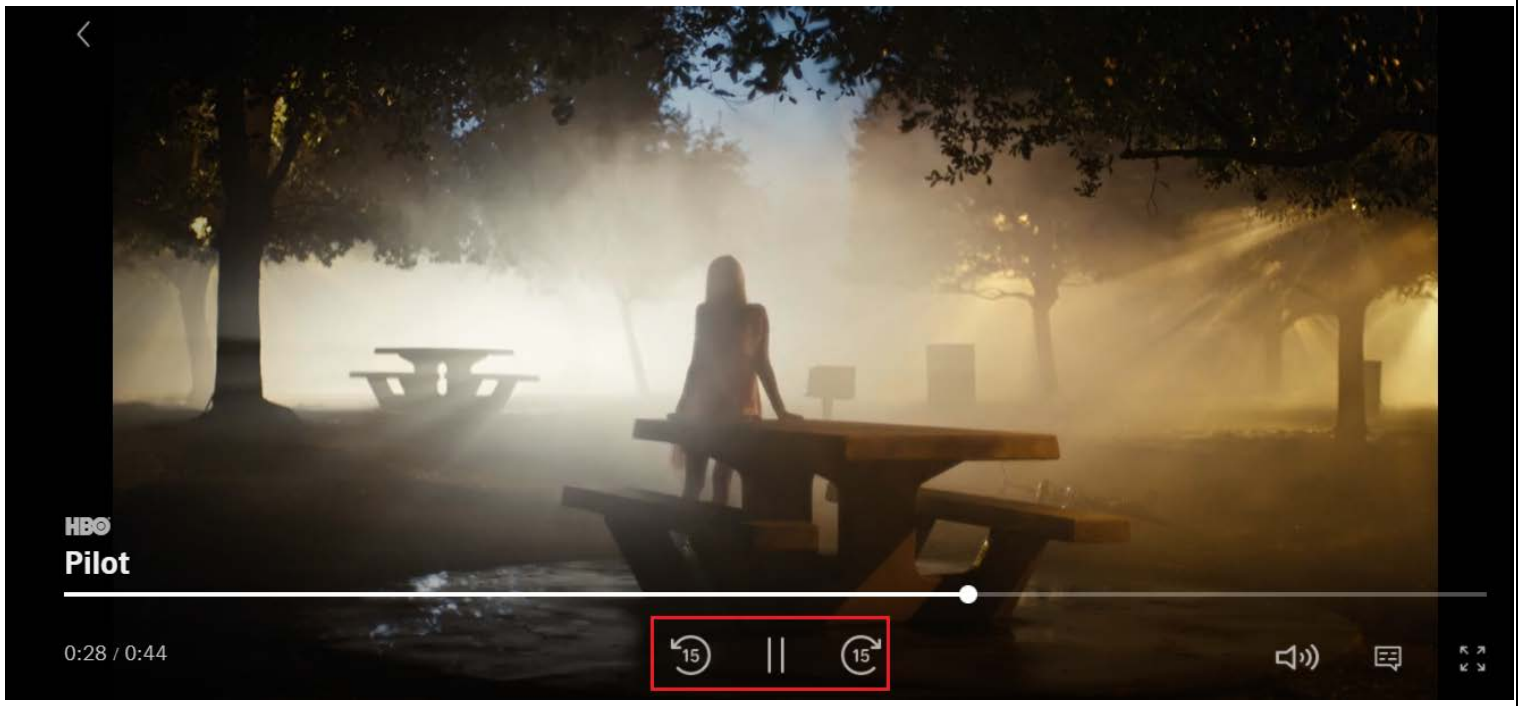
Shown below is the URL of a .mpd master file sent by HBO which identifies the usage of MPEG-DASH-based streaming by HBO. The .mpd master file refers to all the variants of the video encoded for various bandwidths and resolutions. The URL of .mpd master file is:
https://cmaf.lv3.us.hbomaxcdn.com/videos/GYNNHNwDfl6yYnQEAAAAZ/0/00077c/0_JvEEZg_noanc_aBToBqQ_1080hi.mpd

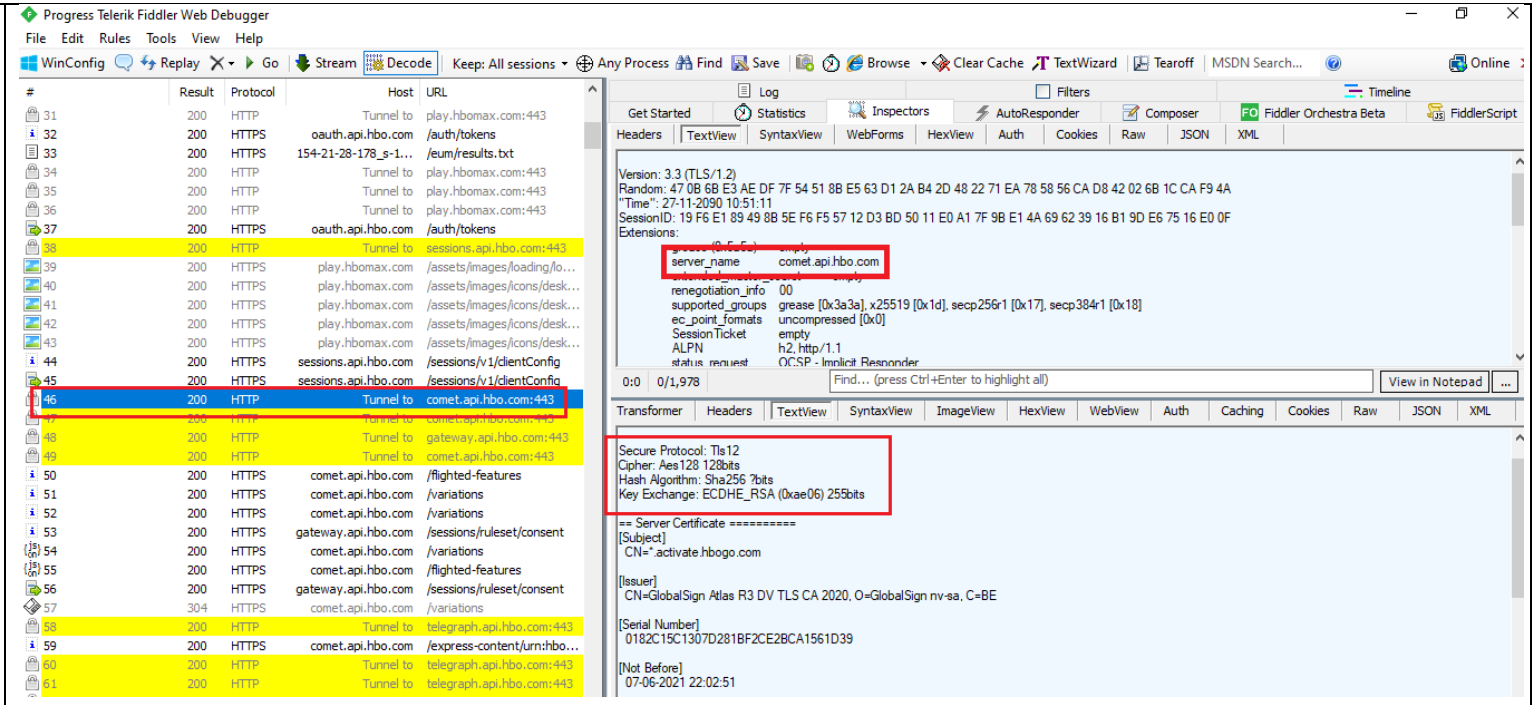


Source: Packet captures by Fiddler

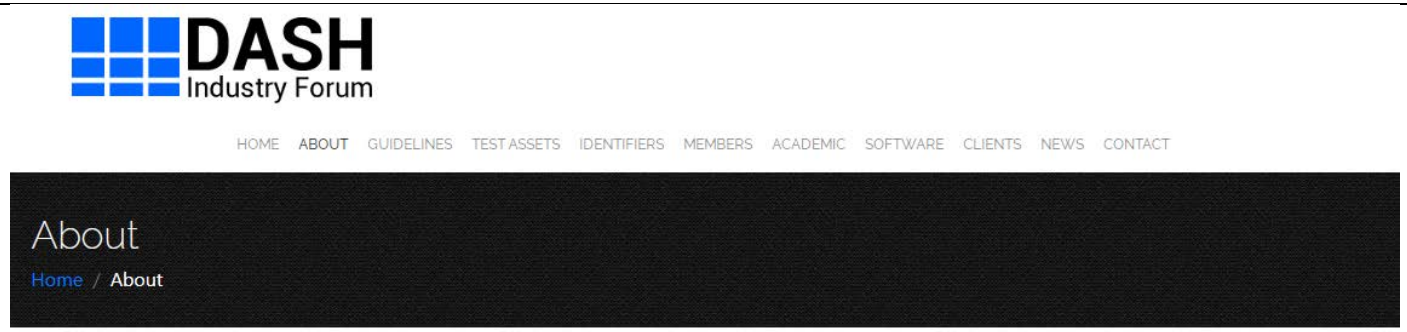


Source: Packet captures by Fiddler

	HBO, as depicted below, provides trick mode operation (such as 15 sec forward/rewind, etc.) to the streamed video.  https://play.hbomax.com/page/urn:hbo:page:GXKN_xQX5csPDwwEAAABj:type:series The Standard practices a process for recording (e.g., recording by means of downloading in a storage), on a recording medium (e.g., a storage mechanism), a scrambled digital video stream (e.g., scrambled video created by utilizing AES 128 encryption), implementing the following steps, in addition to the recording of the scrambled data.



Source: Packet captures by Fiddler



In April 2012, ISO, the international standards body which had already given us the core media foundations of MPEG-2, MP3 and MP4, finally ratified the version of its next generation adaptive streaming standard: MPEG-DASH. In an industry besieged by three comparable (but incompatible) segmented formats many asked – why another? The participating companies in the MPEG-DASH standardization (including Microsoft, Apple, Netflix, Qualcomm, Ericsson, Samsung, and many others) saw a vision of interoperability and convergence required for large-scale market growth that trumped the proprietary and competing solutions. They replaced multiple corporation-controlled solutions with a single industry-defined open standard.

In the same spirit of cooperation in which MPEG-DASH was created, the leading streaming companies got together to form an industry forum to promote and catalyze the adoption of MPEG-DASH and help transition it from a specification into a real business. The DASH Industry Forum (DASH-IF) grew out of a grassroots DASH Promoters Group and was formally incorporated in September 2012. Today it has 67 members spread throughout the world. DASH-IF is filled with member companies who are realists about the DASH deployment challenges. DASH by itself is no magic panacea for the fragmentation problems of media, devices and markets. However, the DASH-IF members do share a common vision that the long-term benefits of convergence outweigh the costs of shorter-term efforts involved in achieving that goal. They are willing to take on the work of creating recommendations, filing bugs, and attending plug-fests

https://dashif.org/about/

	The MPEG-DASH standard is attractive to many companies because of some key market benefits that it brings: • *independent stable international standard* – not owned by any single company, DASH is a finalized specification and not a moving target. • *multi-video and audio tracks* – deliver the complexity of a DVD or Blu-ray experience, with multiple synchronized video and audio options. • *mix of multiplexed and non-multiplexed video and audio tracks* – provide for dynamic bandwidth adaptation, support for multiple audio options such as language selection and surround sound, bandwidth efficiency (sending only the requested tracks) and reduced production, storage, maintenance and delivery costs. • *common encryption* – one-time encryption and packaging of content allowing simultaneous use of multiple DRM technologies. • *templated manifests* – Compact manifest for fast start-up, as well as avoiding manifest download with every segment. • *non-segmented origin files* – files can optionally be stored contiguously on the server and the segments accessed via byte-range requests. • *efficient ad insertion* – server-based and client-based targeted ad-insertion through the use of periods. • *support for multiple CDNs/caches with the same manifest* – provide flexibility to define multiple base URL in the manifest, thus improving scalability and fault tolerance https://dashif.org/about/

4.3.4. Client Operation, Requirements and Guidelines

4.3.4.1. Basic Operation for Single Period

A DASH client is guided by the information provided in the MPD. A simple client model is shown in Figure 5.

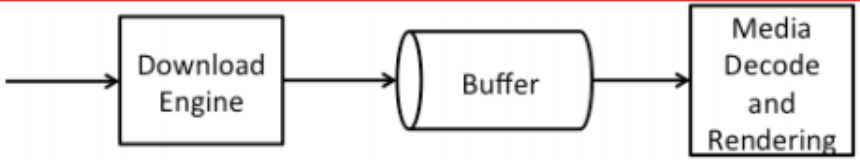


Figure 5 Simple Client Model

Assume that the client has access to an MPD and the MPD contains the parameters in Table 6, i.e. it consumes a dynamic service with fixed media presentation duration.

In addition in the following for simplicity it is assumed that the MPD only contains a single Period with period start time *PSwc*[*i*] and the MPD-URL does not include any fragment parameters according to section 4.3.3.5.

The following example client behavior may provide a continuous streaming experience to the user:

1) The client parses the MPD, selects a collection of Adaptation Sets suitable for its environment based on information provided in each of the **`AdaptationSet`** elements.
2) Within each Adaptation Set it selects one Representation, typically based on the value of the `@bandwidth` attribute, but also taking into account client decoding and rendering capabilities.
3) The client creates a list of accessible Segments at least for each selected Representation taking into account the information in the MPD as documented in Table 6 and the current time *JOIN* in the client and in particular the segment closest to the live edge referred to the *live edge segment*. For details refer to section 4.3.4.2.
4) The client downloads the initialization segment of the selected Representations and then accesses the content by requesting entire Segments or byte ranges of Segments. Typically at any time the client downloads the next segment at the larger of the two: (i) completion of download of current segment or (ii) the Segment Availability Start Time of the next

https://dashif.org/docs/DASH-IF-IOP-v3.3-diff-v3.2.pdf

	7 **7. Content Protection and Security** 8 **7.1. Introduction** 9 DASH-IF IOPs do not intend to specify a full end-to-end DRM system. However DASH-IF IOP 10 provides a framework for multiple DRMs to protect DASH content by adding instructions or *Pro-* 11 *tection System Specific*, proprietary information in predetermined locations in MPDs, or DASH 12 content that is encrypted with Common Encryption as defined in ISO/IEC 23001-7 [31]. 13 The Common Encryption ('cenc') protection scheme specifies encryption parameters that can be 14 applied by a scrambling system and key mapping methods using a common key identifier (KID) 15 to be used by different DRM systems such that the same encrypted version of a file can be com- 16 bined with different DRM systems that can store proprietary information for licensing and key 17 retrieval in the Protection System Specific Header Box ('pssh'), or in **ContentProtection** 18 Descriptors in an MPD. The DRM scheme for each pssh is identified by a DRM specific Sys- 19 temID. https://dashif.org/docs/DASH-IF-IOP-v3.2.pdf Common Encryption [CENC] specifies several protection schemes which can be applied by a scrambling system and used by different DRM systems. The same encrypted DASH presentation can be decrypted by different DRM systems if a DASH client is provided the DRM system configuration for each DRM system, either in the MPD or at runtime. https://dashif-documents.azurewebsites.net/Guidelines-Security/master/Guidelines-Security.html
descrambling of said scrambled data of said stream so as to extract therefrom additional data corresponding to information required by at	The Standard practices descrambling (e.g., decrypting encrypted video created by utilizing common scrambling algorithm) of said scrambled data of said stream (e.g., scrambled video created by utilizing common scrambling algorithm) so as to extract therefrom additional data corresponding to information required by a function of the special mode or "trick mode" (e.g., trick mode adaptation sets/representations).

least one function of the special mode or "trick mode" (fast forward, fast rewind, accelerated motion, slow motion, etc.); and

7. Content Protection and Security

7.1. Introduction

DASH-IF IOPs do not intend to specify a full end-to-end DRM system. However DASH-IF IOP provides a framework for multiple DRMs to protect DASH content by adding instructions or *Protection System Specific*, proprietary information in predetermined locations in MPDs, or DASH content that is encrypted with Common Encryption as defined in ISO/IEC 23001-7 [31].
The Common Encryption ('cenc') protection scheme specifies encryption parameters that can be applied by a scrambling system and key mapping methods using a common key identifier (KID) to be used by different DRM systems such that the same encrypted version of a file can be combined with different DRM systems that can store proprietary information for licensing and key retrieval in the Protection System Specific Header Box ('pssh'), or in **ContentProtection** Descriptors in an MPD. The DRM scheme for each pssh is identified by a DRM specific SystemID.

https://dashif.org/docs/DASH-IF-IOP-v3.2.pdf

Common Encryption [CENC] specifies several protection schemes which can be applied by a scrambling system and used by different DRM systems. The same encrypted DASH presentation can be decrypted by different DRM systems if a DASH client is provided the DRM system configuration for each DRM system, either in the MPD or at runtime.

§ 7. Content encryption and DRM

A DASH presentation MAY provide some or all adaptation sets in encrypted form, requiring the use of a DRM system to decrypt the content for playback. The duty of a DRM system is to prevent disclosure of the content key and misuse of the decrypted content (e.g. recording via screen capture software) and may be to decrypt content.

In a DASH presentation, every representation in an adaptation set SHALL be protected using the same content key (identified by the same `default_KID`).

§ 9. Encryption and DRM signaling in the MPD

A DASH client needs to recognize encrypted content and activate a suitable DRM system, configuring it to decrypt content. The MPD informs a DASH client of the protection scheme used to protect content, identifies the content keys that are used and optionally provides the default DRM system configuration for a set of DRM systems.

The ***DRM system configuration*** is the complete data set required for a DASH client to activate a single DRM system and configure it to decrypt content using a single content key. **It is supplied by a combination of XML elements in the MPD and/or solution-specific logic and configuration**. The DRM system configuration often contains:

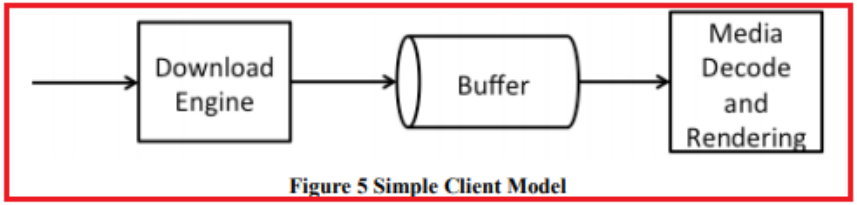
https://dashif-documents.azurewebsites.net/Guidelines-Security/master/Guidelines-Security.html

3.2.9. Trick Mode Support

Trick Modes are used by DASH clients in order to support fast forward, seek, rewind and other operations in which typically the media, especially video, is displayed in a speed other than the normal playout speed. In order to support such operations, it is recommended that the content author adds Representations at lower frame rates in order to support faster playout with the same decoding and rendering capabilities.

However, Representations targeted for trick modes are typically not be suitable for regular playout. If the content author wants to explicitly signal that a Representation is only suitable for trick mode cases, but not for regular playout, it the following is recommended:

- add an Adaptation Set that that only contains trick modes Representations
- annotate the Adaptation Set with an **`EssentialProperty`** descriptor or **`SupplementalProperty`** descriptor with URI `"http://dashif.org/guidelines/trickmode"` and the `@value` the value of `@id` attribute of the Adaptation Set to which these trick mode Representations belong. The trick mode Representations must be time-aligned with the Representations in the main Adaptation Set. The value may also be a white-space separated list of @id values. In this case the trick mode Adaptation Set is associated to all Adaptation Sets with the values of the `@id`.
- signal the playout capabilities with the attribute `@maxPlayoutRate` for each Representation in order to indicate the accelerated playout that is enabled by the signaled codec profile and level.

	https://dashif.org/docs/DASH-IF-IOP-v3.1.pdf
recording of these additional data on the recording medium.	The Standard practices recording of these additional data on the recording medium (e.g., downloading/buffering trick mode adaptations/representations for playback). The product must store the data pertaining to trick modes to allow the playback of the video in trick modes. **4.3.4. Client Operation, Requirements and Guidelines** **4.3.4.1. Basic Operation for Single Period** A DASH client is guided by the information provided in the MPD. A simple client model is shown in Figure 5.  **Figure 5 Simple Client Model** Assume that the client has access to an MPD and the MPD contains the parameters in Table 6, i.e. it consumes a dynamic service with fixed media presentation duration. In addition in the following for simplicity it is assumed that the MPD only contains a single Period with period start time *PSwc*[*i*] and the MPD-URL does not include any fragment parameters according to section 4.3.3.5. The following example client behavior may provide a continuous streaming experience to the user: 1) The client parses the MPD, selects a collection of Adaptation Sets suitable for its environment based on information provided in each of the **`AdaptationSet`** elements. 2) Within each Adaptation Set it selects one Representation, typically based on the value of the `@bandwidth` attribute, but also taking into account client decoding and rendering capabilities. 3) The client creates a list of accessible Segments at least for each selected Representation taking into account the information in the MPD as documented in Table 6 and the current time *JOIN* in the client and in particular the segment closest to the live edge referred to the *live edge segment*. For details refer to section 4.3.4.2.

4) The client downloads the initialization segment of the selected Representations and then accesses the content by requesting entire Segments or byte ranges of Segments. Typically at any time the client downloads the next segment at the larger of the two: (i) completion of download of current segment or (ii) the Segment Availability Start Time of the next segment. Note that if the `@availabilityTimeOffset` is present, then the segments may be downloaded earlier, namely at the adjusted segment availability start time. Based on the buffer fullness and other criteria, rate adaptation is considered. Typically the first media segment that is downloaded is the *live edge segment*, but other decisions may be taken in order to minimize start-up latency. For details on initial buffering, refer to section 4.3.4.4.
5) According to Figure 5 media is fed into buffer and at some point in time, the decoding and rendering of the media is kicked off. The downloading and presentation is done for the selected Representation of each selected Adaptation. The synchronization is done using the presentation time in the Period as documented in section 4.3.2.2.9. For synchronized play-out, the exact presentation times in the media shall be used.

Once presentation has started, the playout process is continuous. The playout process expects media to be present in the buffer continuously. If the **MPD**`@suggestedPresentationDelay` is present, then this value may be used as the presentation delay *PD*. If the **MPD**`@suggestedPresentationDelay` is not present, but the client is expected to consume the service at the live edge, then a suitable presentation delay should be selected, typically between the value of `@minBufferTime` and the value of `@timeShiftBufferDepth`. It is recommended that the client starts rendering the first sample of the downloaded media segment k with earliest presentation time EPT(k) at $PSwc[i] + (EPT(k) - o[r,i]) + PD$. For details on selecting and minimizing end-to-end latency as well as the start-up latency, see section 4.3.4.4.

https://dashif.org/docs/DASH-IF-IOP-v3.3-diff-v3.2.pdf

3.2.9. Trick Mode Support

Trick Modes are used by DASH clients in order to support fast forward, seek, rewind and other operations in which typically the media, especially video, is displayed in a speed other than the normal playout speed. In order to support such operations, it is recommended that the content author adds Representations at lower frame rates in order to support faster playout with the same decoding and rendering capabilities.

However, Representations targeted for trick modes are typically not be suitable for regular playout. If the content author wants to explicitly signal that a Representation is only suitable for trick mode cases, but not for regular playout, it the following is recommended:

- add an Adaptation Set that that only contains trick modes Representations
- annotate the Adaptation Set with an **`EssentialProperty`** descriptor or **`SupplementalProperty`** descriptor with URI `"http://dashif.org/guidelines/trickmode"` and the `@value` the value of `@id` attribute of the Adaptation Set to which these trick mode Representations belong. The trick mode Representations must be time-aligned with the Representations in the main Adaptation Set. The value may also be a white-space separated list of @id values. In this case the trick mode Adaptation Set is associated to all Adaptation Sets with the values of the `@id`.
- signal the playout capabilities with the attribute `@maxPlayoutRate` for each Representation in order to indicate the accelerated playout that is enabled by the signaled codec profile and level.

https://dashif.org/docs/DASH-IF-IOP-v3.1.pdf